

Import Interface Basics

The EMC Import Engine uses an HTML form which must be posted to the following URL:
https://eadvantage.siemens.com/ieg/2_0/import.html

The form data must be "multipart/form-data" encoded.

The HTML form must contain the following form fields:

Field name	Description
iegUsername	The UTF-8 encoded username of the EMC Device User. An EMC Device User can be created by logging into the EMC web front-end and browsing to "User Administration : My Devices : New Device User".
iegPassword	The case-sensitive UTF-8 encoded password of the EMC Device User.
iegImportFile	The UTF-8 encoded plain text import file. The file data may optionally be zipped.

The sample HTML fragment below (use <https://eadvantage.siemens.com/ieg/docs/fileUploadIeg2.html> for a live view) demonstrates the form and the input elements required by the EMC Import Engine:

```
<form action="/ieg/2_0/import.html" method="post" id="fileForm" name="fileForm"
enctype="multipart/form-data">
  <b>Username: </b>
  <input type="text" id="iegUsername" name="iegUsername" value="swpDevice"><br />
  <b>Password: </b>
  <input type="password" id="iegPassword" name="iegPassword" value=""><br />
  <b>Import file: </b>
  <input type="file" id="sourceFile" name="swpImportFile"><br />
  <input type="submit" value="Upload...">
</form>
```

For example, if you were to create a new meter, your *iegImportFile* could look like this:

```
command=METER_CREATE delimiter=; optionalColumnList=startValue,maxMeterValue,comment,manualRead
mk123;mk123;01/01/1980 13:00:00;2;118;86;55;10000;TestReading;1
```

If you now were to submit this *iegImportFile* using the *iegUsername* "iegDevice" and the *iegPassword* "iegDevice", the following would be the dump of the MIME Multipart Media Encapsulation portion of the resulting HTTP request:

```
-----7d72691230b00
Content-Disposition: form-data; name="iegUsername"

iegDevice
-----7d72691230b00
Content-Disposition: form-data; name="iegPassword"

iegDevice
-----7d72691230b00
Content-Disposition: form-data; name="iegImportFile"; filename="C:\Temp\myImportFile.txt"
Content-Type: text/plain

command=METER_CREATE delimiter=; optionalColumnList=startValue,maxMeterValue,comment,manualRead
mk123;mk123;01/01/1980 13:00:00;2;118;86;55;10000;TestReading;1
-----7d72691230b00--
```

Upon completion of the *iegImportFile* processing, the EMC Import Engine responds with a plain text status message.

A status message consists of a list of double vertical-bar '|' separated message elements. The square brackets '[' used in the notation below denote optional message elements while curly brackets denote message elements which are always present. The brackets themselves are not part of actual message syntax.

Status Code	Description
SUC01026	Import complete with no errors.
IEG20000	Nothing imported; no import objects could be created from the source file.
IEG11000	Import complete with errors
USR01024	Invalid user credentials
SYS10000	The system encountered an internal error. The actual description returned varies on the error encountered.

- *SUC01026* is returned upon successful *iegImportFile* processing.
- *IEG11000* is returned upon the partial import of the *iegImportFile*. This indicates that the *iegImportFile* contains some unparsable data lines. Detailed information can be downloaded from the EMC System (see paragraph below).
- *USR01024* is returned upon failure to authenticate the EMC Device User, i.e. password and/or login name of the EMC Device User are wrong.
- *SYS10000* is returned when the EMC Import Engine encounters internal errors, i.e. a connection to the database could not be established or the *iegImportFile* could not be written to the file system, etc.

Optional columns returned by the EMC Import Engine are:

- ID of the log file stored by the EMC System.
- Comma "," separated list of row numbers containing unparsable data.

Each import process performed by a given EMC Device User is logged to the EMC System for controlling purposes. The unique identifier of the stored log information is returned to the client, if applicable. This reduces the amount of data returned to the client once the import process has completed. The detailed log information can be found in the EMC web front-end by logging into the EMC web front-end as the Device User owner and browsing to "User Administration : My Devices : Device Log Symbol".

Note: *SYS10000* errors cannot always be logged to the EMC System, i.e. when database connectivity fails or when an invalid *iegImportFile* has been uploaded.

Here is an example of the detailed log for EMC Device User *iegDevice*:

```
Device Status Log for client: iegDevice
[2007-10-24 02:08:51.533] Status: WARNING - Error creating ImportObject: Unparseable date: "118" in row
number 4
MeterDTO: 58431 imported successfully.
MeterDTO: 58432 imported successfully.
Error creating ImportObject: 3 in row number 1000
Error creating ImportObject: Unparseable date: "03/01/1981 030:00" in row number 5569
Error creating ImportObject: Unparseable date: "04/16/1981 14:4500" in row number 10006
MeterReading inserted batch with: 9997 items.
```



Import File Format

The EMC Import Engine takes all commands and data to be processed from an UTF-8 encoded *Import File*.

- The *Import File* is a plain text file that may either consist of <CR><LF> (Carriage-Return Line-Feed, Windows) or <LF> (Line-Feed, Unix) terminated lines. The last line in the *Import File* does not need to be terminated.
- The *Import File* is read and processed line by line.
- Empty lines or lines that entirely consist of white spaces (blanks, tabulators) are always ignored, no matter at which position they occur in the *Import File*.
- Comments are not supported.
- "<" ">" are not allowed for values.
- The first non-empty line in the *Import File* must specify the command to be executed by the EMC Import Engine (*Command Line*). The first directive on this line must be the "command=" directive that may be followed by further optional directives. Multiple directives on the *Command Line* must be delimited by white spaces (blanks, tabulators).
- All subsequent lines (*Data Lines*) must contain the data for the respective previously given command, unless another *Command Line* is detected. For interactive applications it is not advisable to specify more than one *Command Line* per *Import File*.
- The default delimiter for data columns in *Data Lines* is the comma ',' character. The delimiter string can be configured by the "delimiter=" directive on the *Command Line*. It is the programmer's responsibility to ensure that the specified delimiter string does not occur in the data itself. The asterisk '*' and the period '.' characters must not be used in delimiter strings.
- All *Data Lines* must consist of command-specific mandatory data columns. The order of these mandatory data columns is static and cannot be changed. The mandatory data columns may be followed by command-specific optional data columns which have to be declared on the *Command Line* using the "optionalColumnList=" directive. Each optional data column must be declared by specifying its predefined data column name in a comma ',' delimited list. The order of optional data columns in *Data Lines* must match the order in which they are declared.

The general *Command Line* syntax is defined below where mandatory elements are denoted in curly brackets '{ }' and optional elements in square brackets '[]'. The brackets themselves are not part of the actual *Command Line* syntax.

```
command={command name} [delimiter={character}] [optionalColumnList={column name[,column name]}]
```

Here are some examples for *Import Files*:

```
command=METER_CREATE delimiter=; optionalColumnList=startValue,maxMeterValue,comment,manualRead
cq_03_07_13_40_24;cq_03_07_13_40_24;01/01/1980 13:00:00;2;118;86;55;10000;Test Meter;1
```

```
command=METER_READING delimiter=; optionalColumnList=qualityAttribute,description
cq_03_07_13_40_24;01/02/1981 11:00:00;1;33;Test Reading_0
cq_03_07_13_40_24;01/02/1981 11:15:00;2;33;Test Reading_1
cq_03_07_13_40_24;01/02/1981 11:30:00;3;33;Test Reading_2
cq_03_07_13_40_24;01/02/1981 11:45:00;4;33;Test Reading_3
cq_03_07_13_40_24;01/02/1981 12:00:00;5;33;Test Reading_4
cq_03_07_13_40_24;01/02/1981 12:15:00;6;33;Test Reading_5
cq_03_07_13_40_24;01/02/1981 12:30:00;7;33;Test Reading_6
cq_03_07_13_40_24;01/02/1981 12:45:00;8;33;Test Reading_7
cq_03_07_13_40_24;01/02/1981 13:00:00;9;33;Test Reading_8
cq_03_07_13_40_24;01/02/1981 13:15:00;10;33;Test Reading_9
```

command=METER_READING

The METER_READING command submits meter readings to EMC. Meter readings which already exist in the EMC system will only be overwritten when the EMC Device User is entitled to do so (can be configured in the EMC web front-end under "User administration - My Devices - Edit device - Allow to overwrite values"). Otherwise, the submitted meter readings will be ignored.

Note: Any readings sent to EMC for a Meter (Datasource) that does not exist or is invalid in the system will simply be ignored. No error will be sent to the device log.

Data Columns

Column	Column Name	Req./Opt.	Data Type	Data Range	Description
1	datasource	required	String	1-255 characters	Specifies the ID string that uniquely identifies the <i>datasource</i> of the meter.
2	dateTime	required	UTC Timestamp MM/DD/YYYY HH:mm:ss'	not before '01/01/1950 00:00:00'	Specifies the date and time of the meter reading (Reading Date in the EMC web front-end). It must be given in Universal Coordinated Time (UTC).
3	value	required	Float		Specifies the value of the meter reading.
	qualityAttribute	optional	Integer	Bit mask: 1 = Good 2 = No value 4 = Power outage 8 = Modified 16 = Hardware fault 32 = Automatic value 64 = Unknown 4096 = Negative value 8192 = Null value 16384 = Estimated	The <i>qualityAttribute</i> is a bit mask that can be obtained by applying the Boolean OR operation to respective attribute values. For example, to indicate that the value is unknown and estimated, OR the values 64 and 16384, resulting in a <i>qualityAttribute</i> of 16448. Default value: 33 (Good, Automatic).
	description	optional	String	0-1024 characters	An arbitrary description. Default value: None.

Example:

```
command=METER_READING delimiter=; optionalColumnList=qualityAttribute,description
cq_03_07_13_40_24;01/02/1981 11:00:00;1;33;Test Reading_0
cq_03_07_13_40_24;01/02/1981 11:15:00;2;33;Test Reading_1
cq_03_07_13_40_24;01/02/1981 11:30:00;3;33;Test Reading_2
cq_03_07_13_40_24;01/02/1981 11:45:00;4;33;Test Reading_3
cq_03_07_13_40_24;01/02/1981 12:00:00;5;33;Test Reading_4
cq_03_07_13_40_24;01/02/1981 12:15:00;6;33;Test Reading_5
cq_03_07_13_40_24;01/02/1981 12:30:00;7;33;Test Reading_6
cq_03_07_13_40_24;01/02/1981 12:45:00;8;33;Test Reading_7
cq_03_07_13_40_24;01/02/1981 13:00:00;9;33;Test Reading_8
cq_03_07_13_40_24;01/02/1981 13:15:00;10;33;Test Reading_9
```



Voraussetzung zur automatischen Zählerdatenerfassung: Datenerfassungs- und Übertragungseinheit

Die Datenerfassungseinheit (DEE) kann ein eigenständiges Gerät oder Teil eines Gebäudeautomationssystem sein und muss über folgende Funktionen verfügen:

- Kommunikationsschnittstelle zu Zähler und Fühler
- Pufferfunktion bei Stromausfällen
- Interne Uhr mit Datum und Zeit
- Aufzeichnungsintervall für Tagesaufzeichnung
- Interner Speicher für Ablesewerte mit den Daten:
 - Datum, Uhrzeit
 - Zähleradresse (Name)
 - Ablesewert
- Schnittstelle zur Auswertung

Die Datenübertragungseinheit kann in der DEE integriert, ein eigenständiges Gerät oder Teil eines Gebäudeautomationssystem sein und muss über folgende Funktionen verfügen:

- Kommunikationsmöglichkeit über Internet
- Integrierte EMC Schnittstelle



Automatische Zählerdatenerfassung: Schnittstelle zu EMC

Für die automatische Übermittlung der Zählerdaten von der Datenübertragungseinheit oder für die Teilautomatische Datenübermittlung nach EMC wurde eine neutrale Import Engine entwickelt.

Die Import Engine kann über ein HTML Formular, das an eine definierte URL gepostet wird, automatisch Daten einlesen.

Dabei ist für die Adressierung ein <Devicename> und zur Sicherheit ein zugehöriges Passwort anzugeben.

Die eigentlichen Daten sind in einem genormten *.csv – File zu übertragen.

Nachstehend ein Beispiel für den Aufbau des *.csv – Files:

```
command=METER_READING delimiter=;  
Solarertragszaehler_4711;01/02/2010 11:00:00;1516.23  
Solarertragszaehler_4711;01/02/2010 11:15:00;1519.73
```



Automatische Zählerdatenerfassung: Schnittstelle zu EMC

Die Verwendung der Import Engine ist ausführlich dokumentiert.
Die Dokumentation ist auf Anfrage erhältlich.

command=METER_READING

The METER_READING command submits meter readings to EMC. Meter readings which already exist in the EMC system will only be overwritten when the EMC Device User is entitled to do so (can be configured in the EMC web front-end under "User administration - My Devices - Edit device - Allow to overwrite values"). Otherwise, the submitted meter readings will be ignored.

Note: Any readings sent to EMC for a Meter (Datasource) that does not exist or is invalid in the system will simply be ignored. No error will be sent to the device log.

Data Columns

Column	Column Name	Req./Opt.	Data Type	Data Range	Description
1	datasource	required	String	1-255 characters	Specifies the ID string that uniquely identifies the <i>datasource</i> of the meter.
2	dateTime	required	UTC Timestamp 'MM/DD/YYYY HH:mm:ss'	not before '01/01/1950 00:00:00'	Specifies the date and time of the meter reading (Reading Date in the EMC web front-end). It must be given in Universal Coordinated Time (UTC).
3	value	required	Float		Specifies the value of the meter reading.
	qualityAttribute	optional	Integer	Bit mask: 1 = Good 2 = No value 4 = Power outage 8 = Modified 16 = Hardware fault 32 = Automatic value 64 = Unknown 4096 = Negative value 8192 = Null value 16384 = Estimated	The <i>qualityAttribute</i> is a bit mask that can be obtained by applying the Boolean OR operation to respective attribute values. For example, to indicate that the value is unknown and estimated, OR the values 64 and 16384, resulting in a <i>qualityAttribute</i> of 16448. Default value: 33 (Good, Automatic).
	description	optional	String	0-1024 characters	An arbitrary description. Default value: None.



Datenübergabe

- Firmenneutrale Datenübergabe

Die Datenübergabe an den EBH-Server erfolgt durch Einhaltung der dokumentierten Datenstruktur.

Dadurch ist eine „herstellerneutrale“ Anwendung gegeben.

[import engine gesamtdoku.pdf](#)



Datenübergabe

- Festgelegtes Format

allgemeine Syntax:

```
command={command name} [delimiter={character}] [optionalColumnList={column name[, column name]}]
```

Beispiele für Datensätze:

```
command=METER_CREATE delimiter=;  
optionalColumnList=startValue,maxMeterValue,comment,manualRead  
cq_03_07_13_40_24;cq_03_07_13_40_24;01/01/1980 13:00:00;2;118;86;55;10000;Test Meter;1
```

```
command=METER_READING delimiter=; optionalColumnList=qualityAttribute,description  
cq_03_07_13_40_24;01/02/1981 11:00:00;1;33;Test Reading_0  
cq_03_07_13_40_24;01/02/1981 11:15:00;2;33;Test Reading_1  
cq_03_07_13_40_24;01/02/1981 11:30:00;3;33;Test Reading_2  
cq_03_07_13_40_24;01/02/1981 11:45:00;4;33;Test Reading_3  
cq_03_07_13_40_24;01/02/1981 12:00:00;5;33;Test Reading_4  
cq_03_07_13_40_24;01/02/1981 12:15:00;6;33;Test Reading_5  
cq_03_07_13_40_24;01/02/1981 12:30:00;7;33;Test Reading_6  
cq_03_07_13_40_24;01/02/1981 12:45:00;8;33;Test Reading_7  
cq_03_07_13_40_24;01/02/1981 13:00:00;9;33;Test Reading_8  
cq_03_07_13_40_24;01/02/1981 13:15:00;10;33;Test Reading_9
```